

Aplicaciones

Capa de Aplicación

DC - FCEyN - UBA

30 de Octubre de 2014

Menú del día

1 Domain Name System

- Introducción
- Conceptos
- Registros DNS
- Resolución de consultas
- Ejercicio

2 Correo Electrónico

- Introducción
- MIME
- Enviar
- Recibir
- Ejercicio

3 HTTP

- Introducción
- Implementación
- Ejercicio
- Eficiencia

Información básica

DNS

- * Puerto 53
- * Corre sobre UDP
- * RFC 1034 - Concepts and Facilities
- * RFC 1035 - Implementation and Specification
- * No hay mucho en el Peterson.

¿Qué hace y para qué sirve?

- Alguien se podría manejar en el día a día con direcciones ip?
- Establece una relación entre Nombres de Dominio y objetos de la red (ver Peterson).
- Con esto mi abuela puede usar internet.
- Los servicios cambian de máquina. Mi buscador favorito puede estar hoy en 206.52.102.30 y mañana lo mueven a 50.2.40.200

Soluciones

- /etc/hosts.... Un diccionario (dominio,IP). Se actualiza periódicamente. NO ESCALA, PROPENSO A ERRORES, ETC!!!
- DNS (Domain Name Server). Sistema distribuido que recibe un nombre de dominio y responde la dirección IP (En realidad es un sistema de traducción de nombres genérico. No tiene por qué ser solo usado para direcciones IP).

Decisiones de Implementación

- **Arquitectura cliente-servidor:** Ahorramos mantenimiento
- **Distribuido:** Evitamos fallas graves por caídas de nodos
- **Jerarquía de servidores de nombres:** Le da escalabilidad

Nombres de dominios

- Fully Qualified Domain Name (FQDN) vs Non-FQDN
- Siguen jerarquías.
- Ejemplo de FQDN: mail.dc.fcen.uba.ar.
- Ejemplo de Non-FQDN: mail

Componentes de DNS

- **Registros DNS:** Mantenidos por los administradores. Entre todos forman la base de datos DNS.
- **Servidores DNS (primarios y secundarios):** Servidores Autoritativos que tienen los datos originales.
- **Resolvers (servidores no autoritativos):** Dan respuesta a consultas en redes locales.

Registros DNS: ¿Qué son?

Está formado por los siguientes campos:

- nombre de dominio
- TTL
- clase (en general es IN de internet)
- tipo
- valor

Type	Meaning	Value
SOA	Start of Authority	Parameters for this zone
A	IP address of a host	32-Bit integer
MX	Mail exchange	Priority, domain willing to accept e-mail
NS	Name Server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
HINFO	Host description	CPU and OS in ASCII
TXT	Text	Uninterpreted ASCII text

Ejemplos de registros

nombre de dominio	TTL	clase	tipo	valor
www.dc.uba.ar.	600	IN	CNAME	www-1.dc.uba.ar.
www-1.dc.uba.ar.	600	IN	CNAME	dc.uba.ar.
dc.uba.ar.	600	IN	A	157.92.27.21
dc.uba.ar.	600	IN	NS	ns2.uba.ar.
dc.uba.ar.	600	IN	NS	ns-1.dc.uba.ar.
dc.uba.ar.	600	IN	NS	ns-2.dc.uba.ar.
dc.uba.ar.	600	IN	NS	ns1.uba.ar.
ns1.uba.ar.	1451	IN	A	157.92.1.1
ns1.uba.ar.	6847	IN	AAAA	2001:1318:100c:1::1
ns2.uba.ar.	1451	IN	A	157.92.4.1
ns2.uba.ar.	6847	IN	AAAA	2001:1318:100c:4::1
ns-2.dc.uba.ar.	600	IN	A	157.92.27.253

Servidores primarios y secundarios

Puede haber varios servidores secundarios para back-up y balance de carga. Los servidores secundarios (slave) periódicamente actualizan la información del servidor primario.

El registro SOA indica el nombre del servidor primario y brinda información sobre los updates.

- Serial: Sirve para versionar los registros de la zona.
- Refresh: Determina cuán seguido se actualiza la zona desde el servidor primario.
- Retry: Determina cuánto tiempo esperar para volver intentar en caso que falle el pedido del SOA al primario.
- Expire: Indica cuánto tarda en expirar la zona en caso de no obtener respuesta del primario.
- TTL: Es el tiempo de expiración mínimo para todos los registros de la zona.

Registros DNS: Ejemplo de SOA

```
dc.uba.ar.      IN      SOA      ns1.dc.uba.ar.      admines.dc.uba.ar. (
2014052000
4h
1h
4w
2h
)
```

- ns1.dc.uba.ar : servidor DNS primario
- admines.dc.uba.ar : es el mail de contacto con los responsables (notar que no hay @)
- 2014052000 es el serial. Hay que ponerlo con la fecha del día en que hacés una modificación, si no, no se transfieren los cambios de la zona.
- El resto establece parámetros de configuración de tiempo para el traspaso de los archivos entre distintos servidores DNS.

Registros DNS: Ejemplos de MX

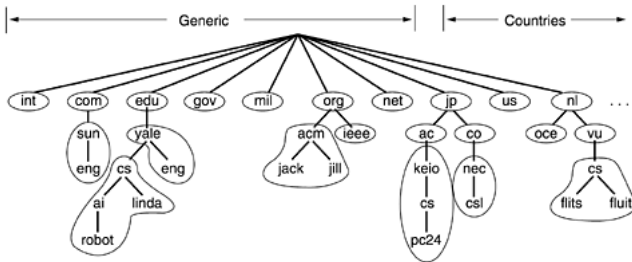
IN	MX	5 mx1.dc.uba.ar.
IN	MX	10 mx1.uba.ar.
IN	MX	20 mx2.uba.ar.

- cuando mandamos mail a qwerty@dc.uba.ar, el servidor final al que se debería mandar el mail es uno de los mencionados arriba.
- Notar que tienen un número antes del nombre, indicando prioridades. A menor el número, es mayor la prioridad.
- Por eso primero se intentaría enviar el mail a mx1.dc.uba.ar, caso contrario a mx1.uba.ar y por último a mx2.uba.ar
- Se asume además de que el servidor autoritativo de dc.uba.ar. sabe al menos la IP de mx1.dc.uba.ar.
- Caso contrario cuando necesite la IP de alguno de los restantes, se lo pregunta al DNS del dominio uba.ar.

Zonas

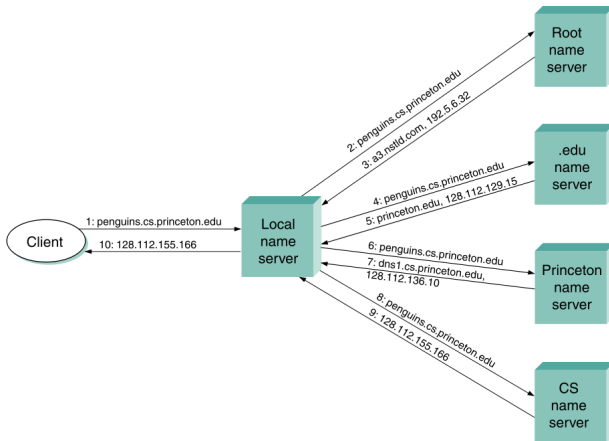
La jerarquía de dominios se separa en zonas:

- Agrupan varios dominios.
- No se solapan.
- Cada zona tiene un name server responsable (un name server puede administrar varias zonas).
- Cada zona se administra por su cuenta.
- En la siguiente slide se entiende mejor la implementación de las zonas y dominios



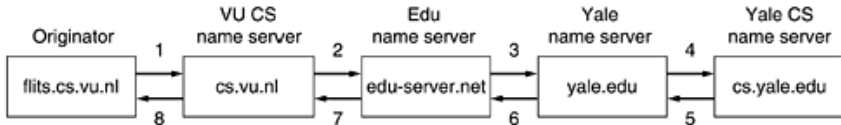
Consultas: Iterativa

Iterativa: cuando un servidor no tiene la dirección responde la dirección del NS que sigue y es responsabilidad del origen hacer la consulta siguiente. Con este metodo se puede cachear los NS que se acceden en el camino.



Consultas: Recursiva

Recursiva: cuando un servidor no tiene la dirección pedida delega la consulta al próximo NS en el camino y cuando recibe la respuesta la envía al origen.



- Los ejemplos antes mostrados son en el peor caso. Es decir, cuando mi resolver local (que podría ser un DNS autoritativo) no conoce el registro buscado ni conoce los servidores autoritativos intermedios.
- Por ejemplo, si mi resolver conocía directamente a los NS de princeton.edu. entonces la iteración empezaría desde ahí y no desde el root.

Respuestas Autoritativas y no Autoritativas

- Si la información viene de un servidor primario o secundario es autoritativa, es decir, está respaldada por el administrador de la zona. En cambio si es información cacheada no se la considera autoritativa y en caso de que no sea correcta se puede recurrir a un servidor de esa zona para obtener una respuesta confiable
- ¿Qué pasa cuando el servidor de una zona esta dentro de esa misma zona? No se puede acceder porque se genera un loop. Solución: Glue records, además del registro NS se agrega el registro A de ese host para que el servidor pueda devolver la IP del NS en cuestión

Ejercicio

`dig` es un comando que permite realizar consultas DNS.

- a. Utilízelo para consultar registros del tipo A, CNAME y MX para los siguientes dominios:
 - `www.dc.uba.ar`
 - `dc.uba.ar`
 - `uba.ar`
 - `ar`
 - `.`
- b. ¿Cómo se reconocen los servidores autoritativos de una determinada zona?
- c. ¿Cómo se distinguen los servidores secundarios del servidor primario?

SMTP, POP3, IMAP, Webmail, MIME



¿Qué partes necesitamos?

- User interface (mail reader).
- Transfer protocol (SMTP).
- Companion protocol (MIME).

Simplicidad del correo

Estrategia

- Los protocolos de correo electrónico son considerablemente simples debido en gran parte a que delegan la interpretación de los datos en otros protocolos o convenciones.
- En los mensajes se transfiere solo texto en formato ASCII.
- La interpretación queda del lado del RFC822 y las extensiones MIME.

Formato de mensaje estándar

- Header (Comandos Separados por CRLF).
 - From:.
 - To:, Cc:, etc.
- Body (ASCII plano).

Extensiones al formato de mensaje

- ¿Por qué?
 - Quiero poder mandar más que solo texto plano.
- ¿Qué permite?
 - Transferir contenido multimedia sin modificar los protocolos básicos.
- ¿Qué se agrega para poder implementarlo?
 - Nuevos campos en el header (MIME-Version:, Content-Type:).
 - Definiciones de tipos y subtipos (Imágenes, Videos, Documentos).
- ¿Cómo hago que todo sea ASCII?
 - Encodings para codificar los distintos tipos de datos en ASCII (base64).
- ¿Cómo hago para enviar más de un tipo de archivo ?
 - Uso delimitadores como técnica de framing.

Ejemplo

```
MIME-Version: 1.0
Content-Type: multipart/mixed;
boundary="-----417CA6E2DE4ABCAFB5"
From: Alice Smith <Alice@cisco.com>
To: Bob@cs.Princeton.edu
Subject: promised material
Date: Mon, 07 Sep 1998 19:45:19 -0400

-----417CA6E2DE4ABCAFB5
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit

Bob,

Here's the jpeg image and draft report I promised.

--Alice

-----417CA6E2DE4ABCAFB5
Content-Type: image/jpeg
Content-Transfer-Encoding: base64
... unreadable encoding of a jpeg figure
```

SMTP

Propiedades esenciales

- La implementación más conocida es sendmail.
- Las casillas de correo son analogías de los buzones.
- Nos podríamos conectar directo con el host donde reside la casilla del destinatario y enviarle el correo con el protocolo SMTP.
- En general nos conectamos con nuestro proveedor y luego el correo pasa por varios gateways antes de llegar al destino.
- ¿Por qué no enviarlo directo al destino?
- Para más información, RFC 2821.

Ejemplo

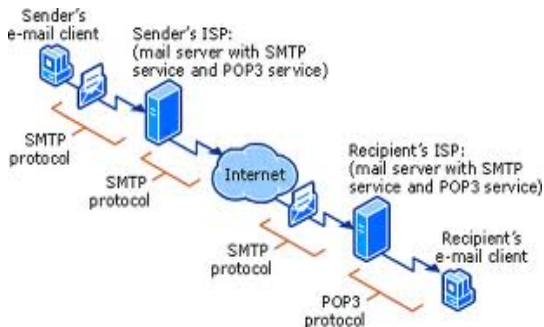
```
S: 220 Servidor ESMTP
C: HELO miequipo.midominio.com
S: 250 Hello, please to meet you
C: MAIL FROM: <yo@midominio.com>
S: 250 Ok
C: RCPT TO: <destinatario@sudominio.com>
S: 250 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: Subject: Campo de asunto
C: From: yo@midominio.com
C: To: destinatario@sudominio.com
C:
C: Hola,
C: Esto es una prueba.
C: Hasta luego.
C:
C: .
S: 250 Ok: queued as 12345
C: quit
S: 221 Bye
```


POP3

Client oriented

- Se descargan los correos y administran localmente.
- Solo permite acceder a la bandeja de entrada.
- No hace falta estar conectado para acceder a los correos.
- El espacio de almacenamiento lo define el usuario.
- Contenido local, ¿Más inseguro?.
- Responsabilidad de backup del lado del usuario.
- Sencillo, fácil de usar y configurar.
- Para más información, RFC 1939.

Ejemplo



POP3

Mensajes

- LIST.
- STAT.
- DELE n.
- QUIT.
- RETR n.
- Sencillo, fácil de usar y configurar.
- Para más información, RFC 1939.

Ejemplo

```
S: <wait for connection on TCP port 110>
C: <open connection>
S: +OK POP3 server ready <1896.697170952@dbc.mtview.ca.us>
C: APOP mrose c4c9334bac560ecc979e58001b3e22fb
S: +OK mrose's maildrop has 2 messages (320 octets)
C: STAT
S: +OK 2 320
C: LIST
S: +OK 2 messages (320 octets)
S: 1 120
S: 2 200
S: .
C: RETR 1
S: +OK 120 octets
S: <the POP3 server sends message 1>
S: .
C: DELE 1
S: +OK message 1 deleted
C: RETR 2
S: +OK 200 octets
S: <the POP3 server sends message 2>
S: .
C: DELE 2
S: +OK message 2 deleted
C: QUIT
S: +OK dewey POP3 server signing off (maildrop empty)
C: <close connection>
S: <wait for next connection>
```

IMAP

Server oriented

- Es el protocolo de acceso más complejo y potente.
- Sirve cuando se accede a la misma casilla desde distintos hosts.
- Permite administrar carpetas del lado del servidor.
- Permite guardar estado.
- Búsqueda del lado del servidor.
- Webmails y celulares lo utilizan.
- ¿Más lento?
- Para más información, RFC 3501.

Ejercicio

SMTP — IMAP vs Webmail vs POP3

- ¿Por qué no se puede recibir correo usando SMTP?
- ¿Qué diferencia hay entre el Webmail, POP3 e IMAP, como soluciones para revisar el correo?

Conceptos fundamentales

Generales

- Usa TCP (puerto 80).
- Un archivo índice con referencia a los diversos objetos.
- Browser y servidor se intercambian mensajes HTTP (HEADER:BODY).
- Cada objeto se referencia por un identificador único (URL).
- HTTP **no** mantiene estado.

Versiones

- HTTP 1.0 es NO persistente.
- HTTP 1.1 sí lo es. Muchos objetos pueden ser transmitidos por una única conexión.

Ejemplo

```
<html>
<head> <title> AMALGAMATED WIDGET, INC. </title> </head>
<body> <h1> Welcome to AWI's Home Page </h1>
 <br>
We are so happy that you have chosen to visit <b> Amalgamated Widget's</b>
home page. We hope <i> you </i> will find all the information you need here.
<p>Below we have links to information about our many fine products.
You can order electronically (by WWW), by telephone, or by fax. </p>
<hr>
<h2> Product information </h2>
<ul>
  <li> <a href="http://widget.com/products/big"> Big widgets </a>
  <li> <a href="http://widget.com/products/little"> Little widgets </a>
</ul>
<h2> Telephone numbers </h2>
<ul>
  <li> By telephone: 1-800-WIDGETS
  <li> By fax: 1-415-765-4321
</ul>
</body>
</html>
```

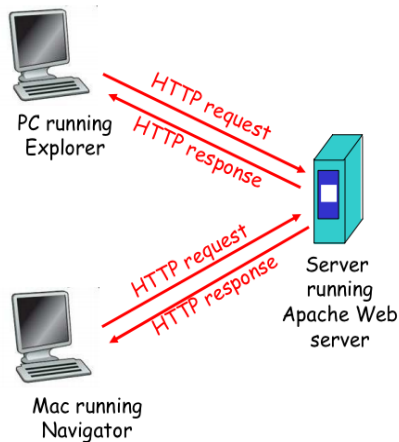
(a)



Modelo HTTP

HTTP: hypertext transfer protocol.

- Protocolo de la capa aplicación de la Web.
- Modelo cliente/servidor
 - *cliente*: browser que requiere, recibe, "despliega" objetos Web.
 - *servidor*: Servidor Web envía objetos en respuesta a requerimientos.
- HTTP 1.0: RFC 1945.
- HTTP 1.1: RFC 2616.



HTTP persistente

¿Por qué?

- $2 * RTT * \text{Objeto}$.
- Muchas conexiones, sobrecarga OS.
- Congestion Window.

Pipelining

- Espera completar los pedidos antes de enviar nuevos requests.
- Con PL, envía request apenas encuentra referencias (default en 1.1).


Petición HTTP

Línea de requerimiento
(comandos GET, POST,
HEAD)

Líneas de encabezado

Carriage return,
line feed

Indica fin de mensaje



```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
User-agent: Mozilla/4.0
Connection: close
Accept-language: es
```

(carriage return, line feed extra)

Tipos de peticiones

HTTP 1.1

- GET.
- HEAD.
- POST.
- PUT.
- DELETE.

Respuesta HTTP

Línea de estatus
(código de estatus
del protocolo
Frase de estatus)

HTTP/1.1 200 OK

Líneas de
encabezado

Connection close

Date: Thu, 06 Aug 1998 12:00:15 GMT

Server: Apache/1.3.0 (Unix)

Last-Modified: Mon, 22 Jun 1998

Content-Length: 6821

Content-Type: text/html

data, e.g.,
archivo
HTML solicitado

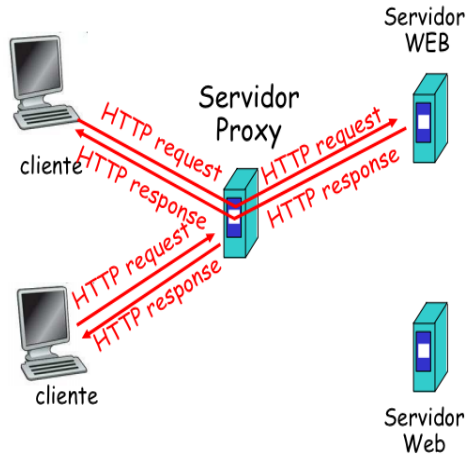
data data data data data ...

Algunos campos del encabezado

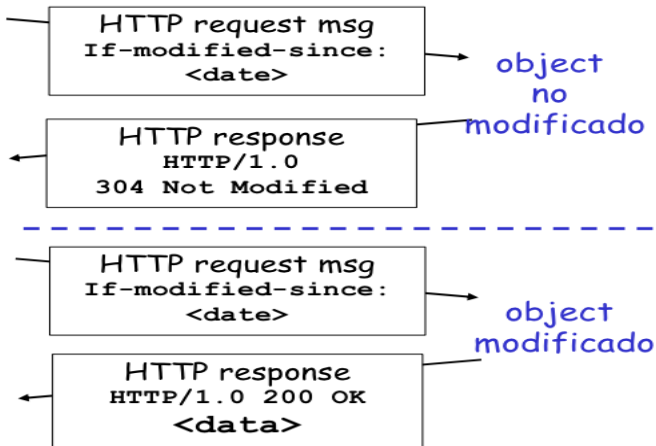
Header	Type	Contents
User-Agent	Request	Information about the browser and its platform
Accept	Request	The type of pages the client can handle
Accept-Charset	Request	The character sets that are acceptable to the client
Accept-Encoding	Request	The page encodings the client can handle
Accept-Language	Request	The natural languages the client can handle
Host	Request	The server's DNS name
Authorization	Request	A list of the client's credentials
Cookie	Request	Sends a previously set cookie back to the server
Date	Both	Date and time the message was sent
Upgrade	Both	The protocol the sender wants to switch to
Server	Response	Information about the server
Content-Encoding	Response	How the content is encoded (e.g., gzip)
Content-Language	Response	The natural language used in the page
Content-Length	Response	The page's length in bytes
Content-Type	Response	The page's MIME type
Last-Modified	Response	Time and date the page was last changed
Location	Response	A command to the client to send its request elsewhere
Accept-Ranges	Response	The server will accept byte range requests
Set-Cookie	Response	The server wants the client to save a cookie

Técnicas de cacheo

- Usuario configura el browser: Acceso Web vía cache.
- Browser envía todos los requerimientos HTTP al cache:
 - Si objeto está en cache: cache retorna objeto.
 - Sino cache requiere los objetos desde el servidor Web, y retorna el objeto al cliente.



Cacheo HTTP



Registros DNS

Dada la siguiente base de registros de un servidor DNS de la facultad de exactas.

Base de registros DNS

exactas.uba.ar. 1w	IN	SOA	exactas.uba.ar jperez.exactas.uba.ar (2005091900 3h 1h 1w 1h)
exactas.uba.ar.	IN	NS	ns.exactas.uba.ar
exactas.uba.ar.	IN	MX	mailserver.exactas.uba.ar
mailserver	IN	CNAME	proxy.exactas.uba.ar
www	IN	A	208.190.1.20
ads	IN	A	208.190.1.21
proxy	IN	A	208.190.1.22
proxy	IN	A	208.190.1.23
proxy	IN	A	208.190.1.24
ns	IN	A	208.190.1.26
zorzal	IN	A	208.190.1.21
pc1	IN	A	208.190.1.30

Consulta DNS

Además de la base de registros, se muestra una salida por pantalla de una consulta DNS:

Consulta parcial

```
;; flags: qr rd ra;  
  
;; QUESTION SECTION:  
;exactas.uba.ar. IN MX  
  
;; ANSWER SECTION:  
....  
  
;; Query time: 3 msec  
;; SERVER: 208.190.1.26#53(208.190.1.26)  
;; WHEN: Sat Feb 31 11:06:56 2046  
;; MSG SIZE rcvd: 328
```

Enunciado

- a. Explique en qué consiste la consulta realizada y complete la sección answer de la consulta con una respuesta válida que permita el acceso al servicio solicitado.
- b. Detalle los mensajes HTTP (*Requests* y *Responses*) y DNS (*Consultas* y *Respuestas*) que desencadena un *Request* GET por el recurso `http://www.exactas.uba.ar/index.html` por parte de la pc1 al servidor proxy con dirección 208.190.1.222 hasta que le llega el respectivo *Response*.

¿Repasando...?

¿Dudas?,
¿Preguntas?